Date=8/12/2020
Lecture By= Akash Handa
Subject ⇒ React Introduction

IN PREVIOUS LECTURE **(QUICK RECAP) Date-7/12/2020**	**In Today's Lecture (Overview)**
React Introduction What is React? How does React Work? React.JS History React ES6 What is ES6? Why Should I Learn ES6? Classes The Render Function Example The HTML Code React JSX What is JSX? Coding JSX React Components React Components Create a Class Component Create a Function Component Component Constructor Example	Passing Functions to Components How do I pass an event handler (like onClick) to a component? How do I bind a function to a component instance? Bind in Constructor (ES2015) Arrow Function in Render Why is binding necessary at all?

Passing Functions to Components

How do I pass an event handler (like onClick) to a component?

Pass event handlers and other functions as props to child components:

```
<button onClick={this.handleClick}>
```

If you need to have access to the parent component in the handler, you also need to bind the function to the component instance (see below).

How do I bind a function to a component instance?

There are several ways to make sure functions have access to component attributes like `this.props` and `this.state`, depending on which syntax and build steps you are using.

Bind in Constructor (ES2015)

```
class Foo extends Component {
  constructor(     ) {
    super(     );
    this.            = this.handleClick.bind(this);
  }
  handleClick() {
           .log('Click happened');
  }
  render() {
    return <button onClick={this.handleClick}>        </button>;
  }}
```

Arrow Function in Render

```
class Foo extends Component {
  handleClick() {
           .log('Click happened');
  }
  render() {
    return <button onClick={() => this.handleClick()}>        </button>;
  }}
```

Why is binding necessary at all?

In JavaScript, these two code snippets are not equivalent:

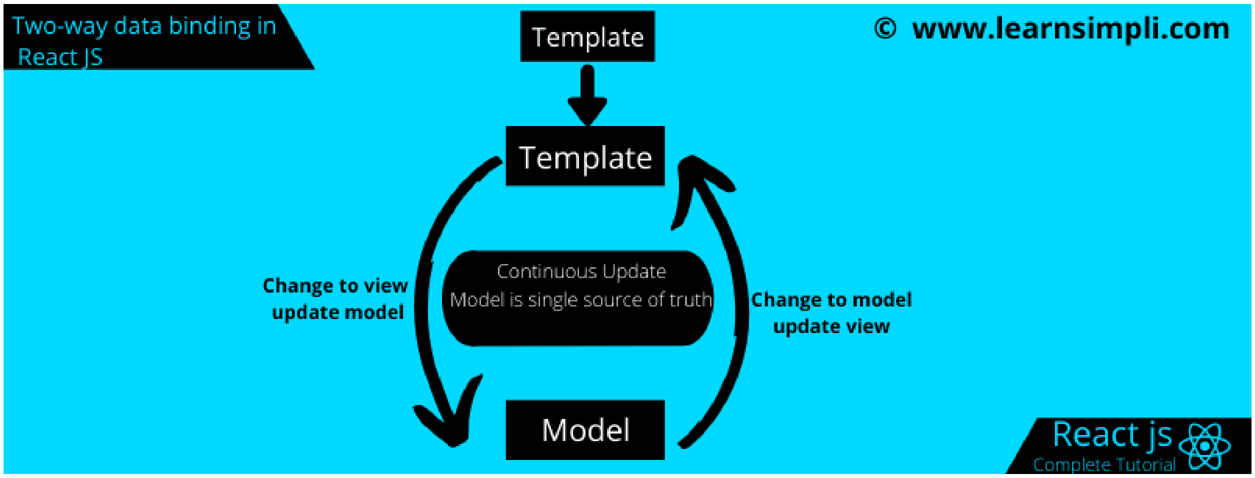
```
obj.method();
```

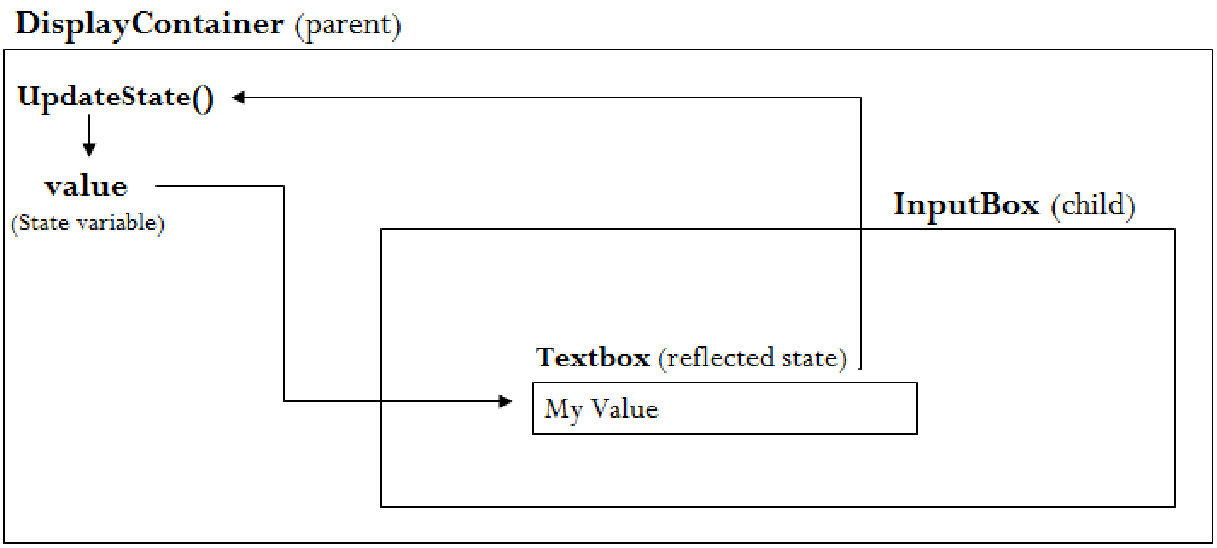
```
var method = obj.method;
```

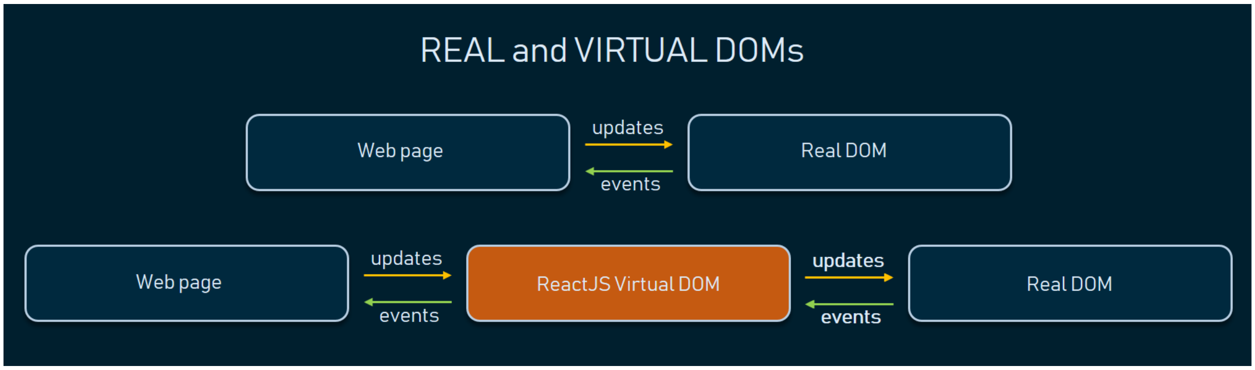
```
method();
```

Binding methods helps ensure that the second snippet works the same way as the first one.

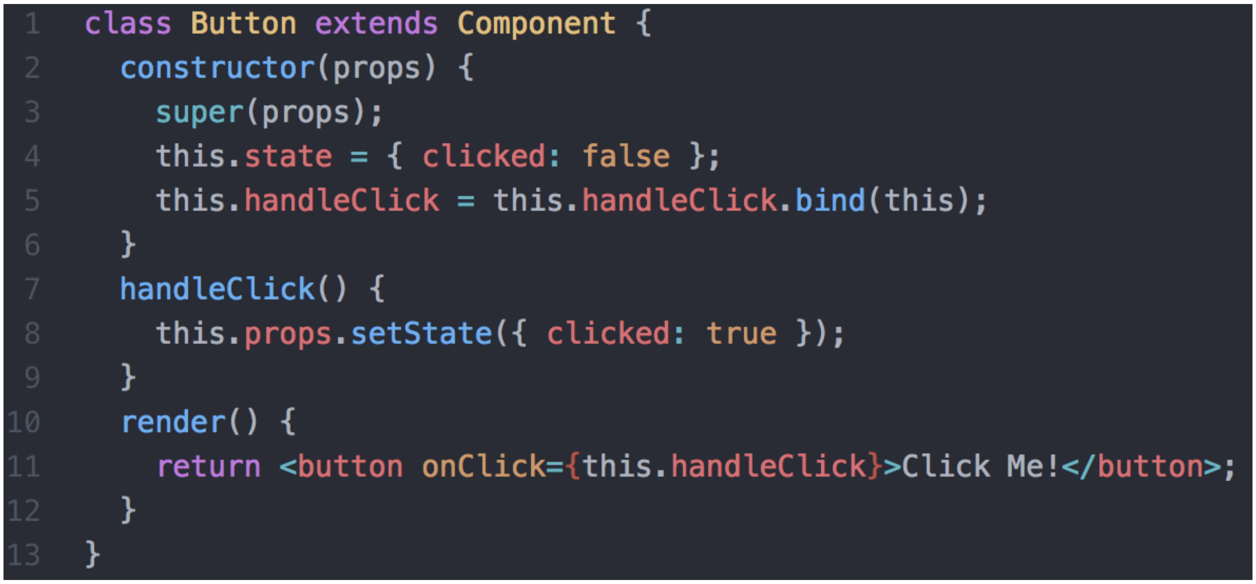
With React, typically you only need to bind the methods you *pass* to other components. For example, `<button onClick={this.handleClick}>` passes `this.handleClick` so you want to bind it. However, it is unnecessary to bind the `render` method or the lifecycle methods: we don't pass them to other components.


Two-way data binding in React JS
Template
© www.learnsimpli.com
Template
Change to view update model
Continuous Update
Model is single source of truth
Change to model update view
Model
React js
Complete Tutorial





A simple Example



```
class Button extends Component {
  constructor(props) {
    super(props);
    this.state = { clicked: false };
    this.handleClick = this.handleClick.bind(this);
  }
  handleClick() {
    this.props.setState({ clicked: true });
  }
  render() {
    return <button onClick={this.handleClick}>Click Me!</button>;
  }
}
```